

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 2000

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

Amendments to the Claims

Please amend claims 1, 6, 13, 15, 20, and 28 as indicated below. This listing of claims will replace all prior versions, and listings, of claims in the application:

1 1. (Currently Amended) An application programming interface (API) for
2 enhancing data network communication, comprising:

3 an identify address function including programming instructions for identifying a
4 stored node address stored by a base driver for a network interface associated with the
5 base driver; and

6 an update node address function including programming instructions for directing
7 the base driver to update the stored node address with a new node address in a
8 configuration storage of the base driver, and in a receive address filtering table stored in
9 [for] the network interface.

10 2. (Original) The API of claim 1, wherein the identify address function
11 includes submitting a request to the base driver, to which is received a response
12 including the node address stored by the base driver.

13 3. (Original) The API of claim 1, wherein the identify address function
14 includes programming instructions for inspecting the configuration storage of the base
15 driver, such storage having an entry identifying the stored node address.

16 4. (Previously Presented) An API according to claim 1, further
17 comprising:

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 a driver identification function including programming instructions for sending an
2 identity-check request to the base driver, said base driver providing a response selected
3 from a group consisting of: a predetermined identifier, a base driver revision number,
4 and an identification of a vendor of the base driver.

5 5. (Original) An API according to claim 4, wherein the predetermined
6 identifier is a copyright string for the vendor of the base driver.

7 6. (Currently Amended) An article of manufacture, comprising a
8 computer readable medium having encoded thereon programming instructions capable
9 of directing a processor to perform operations of:

10 an identify address function for identifying a stored node address stored by a
11 base driver for a network interface associated with the base driver; and

12 an update node address function for directing the base driver to update the
13 stored node address with a new node address in a configuration storage of the base
14 driver, and in a receive address filtering table stored in [for] the network interface.

15 7. (Original) An API according to claim 1, further comprising:
16 a first transmission function including programming instructions for re-transmitting
17 data, received in a compatible format from a network source, in an incompatible format
18 to a network destination; and

19 a second transmission function including programming instructions for re-
20 transmitting data, received in the incompatible format from the network destination, in
21 the compatible format to the network source.

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 8. (Original) An API according to claim 7, further comprising:
2 a report capabilities function including programming instructions for sending the
3 base driver a request to have the base driver report its capabilities;
4 a receive capabilities function including programming instructions for receiving a
5 response including said capabilities;
6 wherein the incompatible format is formatted according to said capabilities.

7 9. (Original) An API according to claim 7, further comprising:
8 a virtual LAN function including programming instructions to direct the base driver
9 to enter a desired virtual LAN operative state; and
10 a disconnect function including programming instructions to notify the base driver
11 that the API has concluded communications with the base driver.

12 10. (Cancelled)

13 11. (Original) An API according to claim 1 for providing transparent fail-
14 over from a first network interface to a second network interface, further comprising:
15 a status function including programming instructions for polling a first base driver
16 for the first network interface to detect a failure of said first network interface;
17 wherein the update node address function includes a function to direct a second
18 base driver for the second network interface to store the node address of the first
19 network interface as the stored node address for the second base driver.

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

12. (Original) An API according to claim 11, in which a Novell ODI compliant network is utilized for network communication, and wherein the update node address function uses at least one ODI MLID Control Routine.

13. (Currently Amended) An article of manufacture, comprising a computer readable medium having encoded thereon instructions to direct a processor to perform an API having:

an identify address function for identifying a stored node address stored by a base driver for a network interface associated with the base driver;

an update node address function for directing the base driver to update the stored node address with a new node address;

a status function in communication with a first base driver for the first network interface to detect a failure of the first network interface; and

a failover function to direct a second base driver for the second network interface to store the node address of the first network interface as the stored node address for the second base driver, and to store the node address of the first network interface in a receive address filtering table stored in [for] the second network interface.

14. (Original) An API according to claim 1 for providing transparent load balancing of data transmissions directed towards the network interface by distributing such data across a second network interface, further comprising:

a queue monitoring function including programming instructions for detecting a workload of the first network interface; and

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 a distribution function including programming instructions for routing a portion of
2 said data transmissions through the second network interface, said distribution function
3 utilizing the update node address function to associate the node identifier of the first
4 network interface with the second network interface.

5 15. (Currently Amended) A networking method for an operational
6 network interface, comprising:

7 receiving first network traffic with a protocol stack;
8 sending said first traffic to an intermediary layer;
9 routing said first traffic to a virtual interface driver;
10 repackaging said first traffic by the virtual interface driver, and providing said
11 repackaged traffic to a virtual protocol stack;
12 sending said repackaged traffic to the intermediary layer;
13 routing said repackaged traffic by the intermediary layer to an interface driver for
14 a network interface having a node address memory stored within the operational
15 network interface;
16 identifying a failed network interface having a node address; and
17 storing the node address in the node address memory.

18 16. (Previously Presented) A method according to claim 15, further
19 comprising:

20 routing network traffic for the failed network interface through the fail over
21 network interface.

22 17. (Original) A method according to claim 16, further comprising:

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 wherein said first network traffic is received in a first protocol format, and said
2 repackaged traffic is in a second network protocol format different from the first protocol
3 format.

4 18. (Previously Presented) A method according to claim 16, wherein locating
5 the fail over network interface comprises:

6 submitting a node identification request to a base driver for a potential fail over
7 network interface; and

8 receiving a response from said driver, said response including an authentication
9 string;

10 verifying said authentication string has a predetermined value before said
11 potential fail over network interface is used as the fail over network interface.

12 19. (Previously Presented) An article of manufacture, comprising a
13 computer readable medium having encoded thereon instructions to direct a processor to
14 perform the operations of:

15 receiving first network traffic with a protocol stack;

16 sending said first traffic to an intermediary layer;

17 routing said first traffic to a virtual interface driver;

18 repackaging said first traffic by the virtual interface driver, and providing said
19 repackaged traffic to a virtual protocol stack;

20 sending said repackaged traffic to the intermediary layer;

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 routing said repackaged traffic by the intermediary layer to an interface driver for
2 a network interface having a node address memory stored within the operational
3 network interface;
4 identifying a failed network interface having a node address; and
5 storing the node address in the node address memory.

6 20. (Currently Amended) A method for redundant networking in a
7 network environment, comprising:
8 determining operative status of a first network interface having a first driver, and
9 of a second network interface having a second driver with a driver memory for storing a
10 MAC address for said second interface;
11 if the first network interface is inoperative, instructing the second driver to store
12 the first network interface MAC address in the driver memory and in a receive address
13 filtering table stored in the second network interface to allow processing by the second
14 network interface of network traffic bound for the first network interface;
15 directing the second driver to activate the second network interface; and
16 directing the first driver to deactivate the first network interface.

17 21. (Original) A method according to claim 20, in which the network
18 environment is a Novell based network, and wherein ODI commands are issued to said
19 first and second drivers.

20 22. (Original) A method according to claim 21, further comprising:
21 receiving first network traffic by a protocol stack;
22 forwarding said first network traffic to a LSL;

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 routing said first network traffic from the LSL to a virtual MLID, and deriving
2 second network traffic from said first network traffic;
3 providing said second network traffic to a virtual protocol stack; and
4 forwarding said second network traffic to the LSL.

5 23-25. (Cancelled)

6 26. (Original) A system, comprising:
7 means for identifying a stored node address stored by a base driver for a network
8 interface associated with the base driver; and
9 means for directing the base driver to update the stored node address with a new
10 node address.

11 27. (Original) A system according to claim 26, further comprising:
12 means for re-transmitting data, received in a first format from a network source,
13 in a second format to a network destination; and
14 means for re-transmitting data, received in the second format from the network
15 destination, in the first format to the network source.

16 28. (Currently Amended) An method for enhancing data network
17 communication comprising:
18 receiving network traffic for a network interface having a first node address;
19 updating a stored node address stored in a receive address filtering table stored
20 in [for] a second network interface, and in a base driver for the second network
21 interface, with the first node address; and

Application No. 09/443,026
Amendment dated July 12, 2004
Response to Office Action of February 12, 200

Atty. Docket No. 042390.P7279
Examiner Charles Anya
TC/A.U. 2126

1 routing the received network traffic to the second network interface.

2 29. (Original) The method of claim 28, wherein said receiving network
3 traffic is performed by an intermediary configured to determine unavailability of the first
4 network interface and automatically update the stored node address of the second
5 network interface filtering table and its base driver so that the second network interface
6 may transparently operate as if it were the first network interface.